ITRONIX®

| TECHNICAL NOTE | Number | 526 |
|---|---|---|
| DATE: 14-Jul-00 | Issue | 1 |

| | |
|---|---|
| **TITLE:** | FS3 BIOS powersave features |
| **ABSTRACT:** | Contains information on using BIOS power management features in applications software for the FS3. |

## Introduction

It has been observed that whilst the Husky FS/2 computer will normally get 8 hours or more out of a set of fully charged batteries in many situations an FS3 running exactly the same application will only get between 4 and 5 hours.

This is most likely to be due to the application keeping the FS3 continuously in full running mode, and not allowing it to use power save modes. In running mode the FS3 draws 320mA or so and with a 1600mAh NiCd this equates to 5 hours at best. This method of operation was not a problem on FS/2, which has a running current of some 110mA and a 1400mAh battery, easily giving a full working day between charges even if the FS/2 never reverts to power-save.

In light of this it is now very important for system designers to implement their FS3 applications with power saving in mind. This document will provide some key points to look for when making an application power friendly and give brief tips on improving the performance of existing applications to be ported from FS/2 to FS3.

## Power Saving Features

The FS3 computer's BIOS contain a number of features to help reduce the amount of power consumed by the unit and greatly extend the battery life. The three categories given in section 33.1.4 of the FS3 System Developers Guide (SDG) are shown below

| | |
|---|---|
| Time-outs | Used to conserve power after configured periods with no activity. |
| Power save modes | HALT and STOP modes are low power consumption modes which are automatically selected by the BIOS when the unit is waiting for something or inactive respectively. |
| Switchable devices | The backlight, RS232 drivers, Power output, PCMCIA cards, Infrared and Rack interfaces can all be turned off from software in order to conserve power. |

Although most of these features are controlled automatically by the BIOS it will usually be necessary to enable or disable them at certain points in an application and to ensure that your code is written in such a way as to allow the features to work.

Both the Time-outs and the power save modes require the unit to become `idle'. This means that although the processor will be `busy' doing something, e.g. while waiting for a key, it must not do any significant input / output. The use of any of the following services will postpone automatic power saving functions.

- Screen services via int10h or direct writing to video RAM
- Disk services via int13h
- Serial communications via int14h or direct access to UARTs
- BIOS extension services via int15h
- Time and Date services via int1Ah
- DOS services via int21h
- Access to 8253 sound

The information contained in this document is copyright and may not be reproduced, stored in a retrieval system in any form by means of, electronic, mechanical, photocopying or otherwise without prior permission from the copyright holder, Itronix (UK) Limited.

Disclaimer. Whilst every precaution has been taken in the preparation of this document the publisher assumes no responsibility for errors and omissions. Neither is any liability assumed for damage resulting from the use of the information contained within this document.

Itronix (UK) Limited, Eden Road, Walsgrave Triangle Business Park, Coventry CV2 2TB, Telephone Coventry (024) 76 604040.

If your code is written for a compiler which uses `standard library routines' you may not even be aware that these services are being used. Generally any ongoing I/O (keyboard screen and disk) functions will prevent the BIOS from evaluating the unit to as idle.

The BIOS can be instructed to allow power save even if some of these functions are in use by using int15h function AH=8Ch (Get/Set power save entry flag) this is an advanced feature and should only be used with great care.

The instructions contained in the next two sections should be sufficient to get the best power saving out of most applications. Some special cases will be covered it the final section.

## Using the BIOS Automatic Power Saving Features

In order for the BIOS time-outs to be used it is necessary for the unit to have been in STOP mode for the period configured for the individual time-outs.

Any application that needs to use the automatic BIOS power save features (time-outs and power save modes) should enable STOP mode during its initialisation by using int15h function AH=76h (SDG chapter26). This needs to be done since the programmer does not know if STOP mode will be enabled or not when the software is first run. The application default should be to use BX=1, enable unconditional power save. If any inhibitions are required they should be used where and only where they are required.

The application should also set up both the idle time-out (automatic power off) and the backlight time-out during initialisation by using int15h function AX=6301h, BX=0 (disable) to E10h seconds (SDG chapter26) and int10h function AX=6501h, BX=0 (disable) to 12Ch seconds (SDG chapter21) respectively.

## Communications and the BIOS Automatic Power Saving Features

Any access of the serial communications ports (COM1, COM2, IR., Fischer and serial PCMCIA devices) will cause the BIOS to use HALT mode rather than STOP mode. Although this will result in lower ongoing power usage it will keep the processor active in order to be prepared to receive incoming interrupts and consequently prevent time-outs from being used correctly.

To re-enable STOP mode after communications is complete the application should follow these steps.

- The handshaking lines (RTS and DTR) should be deactivated and any response from the other end waited for. This should ensure that the other end knows that communications has finished.

- The receive and transmit buffers should be flushed and the communications shut down (UART interrupts disabled, service routines revectored etc. as appropriate). This will prevent further interrupts and other comms activity from causing the BIOS to go back into HALT mode.

- Power output on the port (if used) should be terminated with int15h function AH=81h, AL=0 and BL=0 for COM1 or BL=1 for COM2 (SDG chapter26). This should be done whenever a device powered by the FS3 has finished being used since the VCCout can supply up to 400mA and even when nothing is connected will cause a drain of up to 50 mA.

- If the Charging and Communications Rack or Infrared port interfaces were used they should be turned off.

The information contained in this document is copyright and may not be reproduced, stored in a retrieval system in any form by means of, electronic, mechanical, photocopying or otherwise without prior permission from the copyright holder, Itronix (UK) Limited.

Disclaimer. Whilst every precaution has been taken in the preparation of this document the publisher assumes no responsibility for errors and omissions. Neither is any liability assumed for damage resulting from the use of the information contained within this document.

Itronix (UK) Limited, Eden Road, Walsgrave Triangle Business Park, Coventry CV2 2TB, Telephone Coventry (024) 76 604040.

- RS232 drivers should be turned off using int14h function AH=25h, AL=0 and DX=0 for COM1 or DX=1 for COM2 (SDG chapter25). (Drivers are automatically reactivated whenever the UARTs are accessed.)

- Finally the application should re-enable STOP mode as described in the previous section.

## Other Situations to look out for

Some applications will have special power requirements not covered above or in the System Developer's Guide but generally the same principals will apply for conserving power. That is, when a feature is not actively in use turn it off.

The two situations below can cause problems if they are not handled very carefully and may require a lot of work to resolve. If battery life is a critical issue it is essential to analyse the power requirements thoroughly to avoid unnecessary or accidental high current consumption.

### PCMCIA Cards

The FS3 may be fitted with an internal PCMCIA device which will require power. The HCARD utility is used to control the card via int2Fh (See Section 31.3 of the System Developer's Guide for further details). When not in use the card should be deselected by using function AX=C603h.

Sometimes an application may need to have the card active for a significant portion of the time, such as in a GPS application where positional information is constantly required. If this is the case then it may be necessary to use the RTC alarm service to periodically read the card (perhaps every few seconds or so). In such a case the current needed to power the card up may be greater than that required to sustain it so the card could be left on but only accessed (see communications section above) at intervals to allow the unit to go in to STOP mode most of the time.

Modem cards are far easier since they will only be used during communications and can be selected and deselected only when comms is required.

All cards and applications should be dealt with individually to get the best from them.

### Using Idle Time

Some applications will use idle time, such as that available whilst waiting for user input, to perform background tasks. These tasks may include things like updating an on screen clock, monitoring or optimising disk usage, checking communications devices, etc. and should be avoided at all costs.

If it is absolutely necessary to perform such tasks they should be done rarely so as to allow the BIOS to spend as much time as possible in STOP mode.

One way to achieve this is to use the RTC alarm to schedule the tasks at regular intervals rather than allowing them to be ongoing during idle time.

## Conclusion

Power saving is an essential part of programming for the FS3 and can often be a complicated issue. As such it deserves great attention during the application design phase and should never be neglected.

The information contained in this document is copyright and may not be reproduced, stored in a retrieval system in any form by means of, electronic, mechanical, photocopying or otherwise without prior permission from the copyright holder, Itronix (UK) Limited.

Disclaimer. Whilst every precaution has been taken in the preparation of this document the publisher assumes no responsibility for errors and omissions. Neither is any liability assumed for damage resulting from the use of the information contained within this document.

Itronix (UK) Limited, Eden Road, Walsgrave Triangle Business Park, Coventry CV2 2TB, Telephone Coventry (024) 76 604040.

| TECHNICAL NOTE | Number | 526 |
|---|---|---|
| DATE: 14-Jul-00 | Issue | 1 |

Page 1

TECHNICAL NOTE

Number
526
DATE: 10-May-00
Issue
1

The information contained in this document is copyright and may not be reproduced, stored in a retrieval system in any form by means
of, electronic, mechanical, photocopying or otherwise without prior permission from the copyright holder, WPI Husky Computers Limited

Disclaimer. Whilst every precaution has been taken in the preparation of this document the publisher assumes no responsibility for errors and omissions. Neither is any liability assumed for damage resulting from the use of the information contained within this document.

WPI Husky Computers Limited, Eden Road, Walsgrave Triangle Business Park, Coventry CV2 2TB. Telephone Coventry (01203) 604040.

The information contained in this document is copyright and may not be reproduced, stored in a retrieval system in any form by means of, electronic, mechanical, photocopying or otherwise without prior permission from the copyright holder, Itronix (UK) Limited.

Disclaimer. Whilst every precaution has been taken in the preparation of this document the publisher assumes no responsibility for errors and omissions. Neither is any liability assumed for damage resulting from the use of the information contained within this document.

Itronix (UK) Limited, Eden Road, Walsgrave Triangle Business Park, Coventry CV2 2TB, Telephone Coventry (024) 76 604040.